TUTORIAL
Vaclav Skala: Mathematical Foundamentals for Computer Graphics and Virtual Reality, Univ.of West Bohemia,Torino, Italy, October 6-8, 2008

Mathematical Foundamentals for Computer Graphics and Virtual Reality

Vaclav Skala
http://herakles.zcu.cz/~skala
skala@kiv.zcu.cz
subject: Intuition 2008 – Tutorial

Center of Computer Graphics and Visualization http://herakles.zcu.cz
Department of Computer Science and Engineering http://www.kiv.zcu.cz
Faculty of Applied Sciences http://www.fav.zcu.cz
University of West Bohemia http://www.zcu.cz

Plzen, Czech Republic

Plzen (Pilsen)

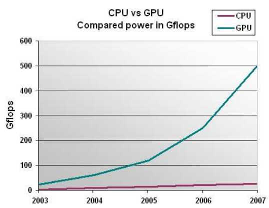

Plzen – is a very old city, city of culture, industry, brewery.
City, where today's beer fermentation was invented
that is why today's beer are called Pilsner

"Real science" in XXI century

Motivation

Technology

- CPU speed, Quad Core architecture
- GPU performance, availability
- Matrix-vector operations
- Memory available 4 GB, disk 2 TB
- Numerical representations



Algorithms stability and robustness

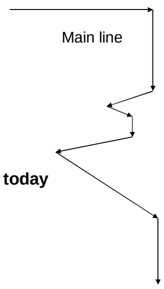
Geometry & Algebra

- From tricks to exact approaches

Contents

History of Geometry

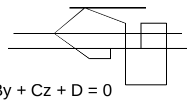
- Euclid - synthetic geometry 300 BC
- Descartes - analytic geometry 1637
- Gauss – complex algebra 1798
- Hamilton – quaternions 1843
- Grassmann – Grasmann Algebra 1844
- Cayley – Matrix Algebra 1854
- Clifford – Clifford algebra 1878
- **Gibbs – vector calculus 1881 – used today**
- Sylvester – determinants 1878
- Ricci – tensor calculus 1890
- Cartan – differential forms 1908
- Dirac, Pauli – spin algebra 1928
- Hestenes – Space-time algebra 1966 ➔ **Geometry Algebra** 1984



Geometry & Computers

- Mathematically perfect algorithms fail due to instability
- Main issues
  - stability, robustness of algorithms
  - acceptable speed
  - linear speedup – results depends on HW, CPU …. parameters !
- Numerical stability
  - limited precision of **float** / **double**
  - tests A **?** B with **float**s
    - **if** A = B **then** ….. **else** ….. ; **if** A = 0 **then** ….. **else** ….
      should be forbidden in programming languages
  - division operation should be removed or postponed to the last moment if possible - "blue screen", system reset etc.

Geometry & Computers



- Typical examples of instability:
  - intersection of 2 lines in E^3,
  - point lies on a line in E^2 or a plane in E^3
    $Ax + By + C = 0$ or $Ax + By + Cz + D = 0$
  - detection if a line intersects a polygon, touches a vertex or passes through

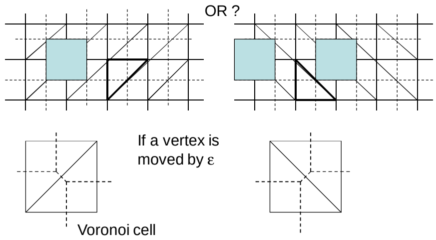
Typical problem

```
double x = -1; double p = ….;
while ( x < +1)
{ if x == p) Console.Out.WriteLine(" *** ")
  x += p;
}
```

if p = 0.1 then no output, if p = 0.25 then expected output

Geometry & Computers

- point inside of a circle given by three points – Delaunay triangulation – problems with meshing points in regular rectangular grid.

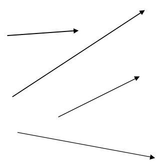


Geometry & Computers

- Precision issues – basic rules
  - [X] + [Y] = [Xmin+Ymin, Xmax+Ymax]
  - [X] - [Y] = [Xmin-Ymax, Xmax-Ymin]
  - [X] * [Y] = [min{Xmin*Ymin, Xmin*Ymax, Xmax*Ymin, Xmax*Ymax}, max{Xmin*Ymin, Xmin*Ymax, Xmax*Ymin, Xmax*Ymax}]
  - [X] / [Y] = (-∞,∞) if Y=0 ; [X] / [Y] = **[X] * [1/X, 1/Y] otherwise !!!!!**
  - √[X] = [√ Xmin, √ Xmax]
- Typical example of wrong computational result:
  $F(x,y)=333.75\, y^6+x^2\,(11x^2y^2-y^6-121y^4-2) +5.5y^8 +x/(2y)$
  at [x,y]=[77 617, 33 096]
  single 6.33 10^29, double 1,172…,
  exact [-0.82739…± 1^{-34}] if interval arithmetic used
  [Lecrerc,A.: Should we be concerned about round-off error?]

Vectors and Points in Geometry

- A vector ***v*** has a magnitude (length) and direction.
- Normalized vectors have magnitude 1, e.g. ‖***v***‖=1
- Zero vector ***0*** has magnitude zero, no direction
- Vectors do not have a location!!
- **Vectors and points have only a similar representation** !!



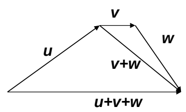
Vectors and Points in Geometry

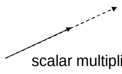
Algebraic rules - properties

- $\boldsymbol{v}+\boldsymbol{w} = \boldsymbol{w}+\boldsymbol{v}$ commutative
- $\boldsymbol{u}+(\boldsymbol{v}+\boldsymbol{w}) = (\boldsymbol{u}+\boldsymbol{v})+\boldsymbol{w}$ associative
- $\boldsymbol{u}+\boldsymbol{0} = \boldsymbol{u}$ additive identity

For every $\boldsymbol{v}$ there is a vector $-\boldsymbol{v}$ such that $\boldsymbol{v}+(-\boldsymbol{v}) = \boldsymbol{0}$

- $(ab)\boldsymbol{v} = a(b\boldsymbol{v})$ associative
- $(a+b)\boldsymbol{v} = a\boldsymbol{v}+b\boldsymbol{v}$ distributive
- $a(\boldsymbol{v}+\boldsymbol{w}) = a\boldsymbol{v}+b\boldsymbol{w}$ distributive
- $1.\boldsymbol{v} = \boldsymbol{v}$ multiplicative identity





scalar multiplication

Vectors and Points in Geometry

Vector representation in R^2

$\boldsymbol{x}_0+\boldsymbol{x}_1 = (x_0,y_0)+(x_1,y_1) = (x_0+x_1,y_0+y_1)$

$a\,\boldsymbol{x}_0 = a(x_0,y_0) = (ax_0,ay_0)$

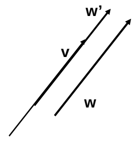
Direct generalization to R^3 and R^n

Linear combination

$\boldsymbol{v} = a_1\boldsymbol{v}_1+\ldots..+ a_n\boldsymbol{v}_n$

If $\boldsymbol{v}_i = a_1\boldsymbol{v}_1+.. a_{i-1}\boldsymbol{v}_{i-1}+ a_{i+1}\boldsymbol{v}_{i+1}+\ldots+ a_n\boldsymbol{v}_n$

then $\boldsymbol{v}_i$ is linearly dependent – two linearly dependent vectors $\boldsymbol{v}$ and $\boldsymbol{w}$ are said to be parallel, e.g. $\boldsymbol{w} = a\boldsymbol{v}$



Vectors and Points in Geometry

Standard vector basis for R^3

$\boldsymbol{e}_1 = (1,0,0),\ \boldsymbol{e}_2 = (0,1,0),\ \boldsymbol{e}_3 = (0,0,1)$

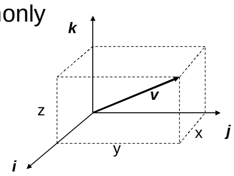
usually as $\{\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}\}$, where

$\boldsymbol{i} = (1,0,0),\ \boldsymbol{j} = (0,1,0),\ \boldsymbol{k} = (0,0,1)$

coefficients a_1, a_2, a_3, commonly represented as x, y, z

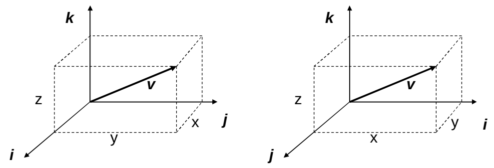
Vector $\boldsymbol{v}$ in R^3

$\boldsymbol{v} = x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k}$



Vectors and Points in Geometry

Right handed R^3 Left handed R^3



! **Be careful** – *column* x *row* vector notation – matrices of geometric transformations are transposed

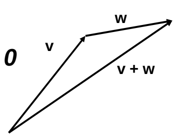
Vectors and Points in Geometry

Vector length and norm

$\|\boldsymbol{v}\| \geq 0$ $\|\boldsymbol{v}\|=0 \Leftrightarrow \boldsymbol{v} = \boldsymbol{0}$

$\|a\,\boldsymbol{v}\| = |a|\,\|\boldsymbol{v}\|$

$\|\boldsymbol{v} + \boldsymbol{w}\| \leq \|\boldsymbol{v}\| + \|\boldsymbol{w}\|$



$\|\boldsymbol{v}\| = (\sum v_i^2)^{1/2}$ Euclidean norm usually Pythagorean theorem $x^2 + y^2 = d^2$

in R^3 $x^2 + y^2 + z^2 = \|\boldsymbol{v}\|^2$

Vectors and Points in Geometry

Inner (dot) product

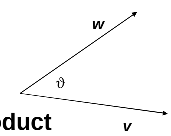
$\langle \boldsymbol{v}, \boldsymbol{w}\rangle = \langle \boldsymbol{w}, \boldsymbol{v}\rangle$ - symmetry

$\langle \boldsymbol{u}+\boldsymbol{v}, \boldsymbol{w}\rangle = \langle \boldsymbol{u}, \boldsymbol{w}\rangle + \langle \boldsymbol{v}, \boldsymbol{w}\rangle$ - additivity

$a\langle \boldsymbol{v}, \boldsymbol{w}\rangle = \langle a\,\boldsymbol{v}, \boldsymbol{w}\rangle = \langle \boldsymbol{v}, a\,\boldsymbol{w}\rangle$ - homogenity

$\langle \boldsymbol{v}, \boldsymbol{v}\rangle \geq 0$ - positivity

$\langle \boldsymbol{v}, \boldsymbol{v}\rangle = 0 \Leftrightarrow \boldsymbol{v} = \boldsymbol{0}$ - definiteness



Euclidean inner product – **dot product**

$\langle \boldsymbol{v}, \boldsymbol{w}\rangle = \boldsymbol{v} \,.\, \boldsymbol{w} = \boldsymbol{v}^T \boldsymbol{w} = \|\boldsymbol{v}\|\,\|\boldsymbol{w}\|\cos\vartheta$

Vectors and Points in Geometry

Cross (vector) product

$\boldsymbol{v} \times \boldsymbol{w} = -(\boldsymbol{w} \times \boldsymbol{v})$ – anti-symmetry - it is not commutative!!

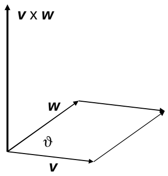
$\boldsymbol{u} \times (\boldsymbol{v} + \boldsymbol{w}) = \boldsymbol{u} \times \boldsymbol{v} + \boldsymbol{u} \times \boldsymbol{w}$

$(\boldsymbol{u} + \boldsymbol{v}) \times \boldsymbol{w} = \boldsymbol{u} \times \boldsymbol{w} + \boldsymbol{v} \times \boldsymbol{w}$

$a(\boldsymbol{u} \times \boldsymbol{v}) = (a\boldsymbol{u}) \times \boldsymbol{v} = \boldsymbol{u} \times (a\boldsymbol{v})$

$\boldsymbol{u} \times \boldsymbol{0} = \boldsymbol{0} \times \boldsymbol{u} = \boldsymbol{0}$

$\boldsymbol{v} \times \boldsymbol{v} = \boldsymbol{0}$



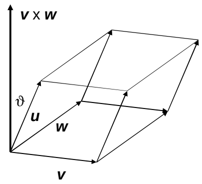
$\|\boldsymbol{v} \times \boldsymbol{w}\| = \|\boldsymbol{v}\| \, \|\boldsymbol{w}\| \sin\vartheta$

The length of the $\boldsymbol{v} \times \boldsymbol{w}$ equals to **area** of parallelogram

Vectors and Points in Geometry

Cross (vector) product

$$\boldsymbol{v} \times \boldsymbol{w} = \det \begin{vmatrix} \boldsymbol{i} & \boldsymbol{j} & \boldsymbol{k} \\ u_x & u_y & u_z \\ v_x & v_y & v_z \end{vmatrix}$$



Vector triple product

$\boldsymbol{u} \times (\boldsymbol{v} \times \boldsymbol{w}) = (\mathbf{u.w})\mathbf{v} - (\mathbf{u.v})\mathbf{w}$

Scalar triple product – equals to the volume of a parallelopiped

$\boldsymbol{u} \,.\, (\boldsymbol{v} \times \boldsymbol{w}) = \|\boldsymbol{u}\| \, \|\boldsymbol{v} \times \boldsymbol{w}\| \cos\vartheta$

Vectors and Points in Geometry

Algebraic rules:

Other vector triple product

$\boldsymbol{u} \times (\boldsymbol{v} \times \boldsymbol{w}) = (\boldsymbol{u.w})\boldsymbol{v} - (\boldsymbol{u.v})\boldsymbol{w}$

$(\boldsymbol{u} \times \boldsymbol{v}) \times \boldsymbol{w} = (\boldsymbol{u.w})\boldsymbol{v} - (\boldsymbol{v.w})\boldsymbol{u}$

$\boldsymbol{u}.(\boldsymbol{v} \times \boldsymbol{w}) = \boldsymbol{w}.(\boldsymbol{u} \times \boldsymbol{v}) = \boldsymbol{v}.(\boldsymbol{w} \times \boldsymbol{u}) = \det[\mathbf{u}|\mathbf{v}|\mathbf{w}]$

$$(\mathbf{a} \times \mathbf{b}).(\mathbf{c} \times \mathbf{d}) = \det \begin{vmatrix} \mathbf{a}^T\mathbf{c} & \mathbf{b}^T\mathbf{c} \\ \mathbf{a}^T\mathbf{d} & \mathbf{b}^T\mathbf{d} \end{vmatrix}$$

$(\mathbf{a} \times \mathbf{b}) \times (\mathbf{c} \times \mathbf{d}) = \mathbf{c}^T[\mathbf{abd}] - \mathbf{d}^T[\mathbf{abc}]$

$= \mathbf{b}^T[\mathbf{acd}] - \mathbf{a}^T[\mathbf{bcd}]$

Vectors and points in geometry

Outer (tensor) product – result is a ***MATRIX Q***

$\boldsymbol{Q} = \boldsymbol{u}\,\boldsymbol{v}^T = \boldsymbol{u} \otimes \mathbf{v}$

$q_{ij} = u_i \, v_j$ - will be defined latter

Useful

$\mathbf{a}^T\mathbf{b} \,.\, \mathbf{c}^T\mathbf{d} = \mathbf{a}^T \,(\mathbf{b} \otimes \mathbf{c})\mathbf{d} = \mathbf{a}^T \mathbf{Q}\, \mathbf{d}$

Tip:

- CPU and GPU optimization for vector/parallel computation if **Q** is a constant

Vectors and points in geometry

Useful formula

$$\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_\times \mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

$$\text{if} \quad \mathbf{a} = \mathbf{c} \times \mathbf{d} \qquad [\mathbf{a}]_\times = \left(\mathbf{cd}^T\right)^T - \mathbf{cd}^T$$

For data visualization

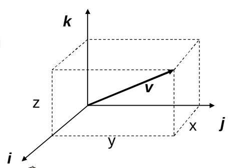
$$\nabla \times (\nabla \times \mathbf{f}) = \nabla(\nabla.\mathbf{f}) - (\nabla.\nabla)\mathbf{f} =$$

$$grad\,(div\,\mathbf{f}) - laplacian\,\mathbf{f}$$

Vectors and Points in Geometry

Vectors – movable, no fixed position

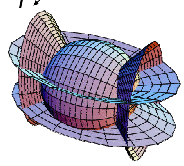
Points – no size, position fixed in the GIVEN coordinate system



Coordinate systems

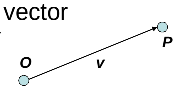
- Cartesian – right handed system is used
- Polar
- Spherical
- and many others

[Confocal Ellipsoidal Coordinates (http://mathworld.wolfram.com/ ConfocalEllipsoidalCoordinates.html)]



Vectors and Points in Geometry

An ***affine space*** is formed by a vector space ***V*** and set of points ***W***



Let us define relations

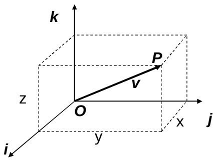
$\boldsymbol{v} = \boldsymbol{P} - \boldsymbol{O}$

$\boldsymbol{P} = \boldsymbol{O} + \boldsymbol{v}$

using *n* basis vectors of ***V***

$\boldsymbol{P} = \boldsymbol{O} + a_1\boldsymbol{v}_1 + a_2\boldsymbol{v}_2 + \ldots + a_n\boldsymbol{v}_n$

Coordinate Frame



Vectors and Points in Geometry

An ***affine combinations***

$\boldsymbol{P} = a_1\boldsymbol{P}_1 + a_2\boldsymbol{P}_2 + \ldots + a_k\boldsymbol{P}_k$

and $a_1 + a_2 + \ldots + a_k = 1$ or $a_1 = 1 - a_2 - \ldots - a_k$

Convex combination $0 \le a_1 + a_2 + \ldots + a_k \le 1$

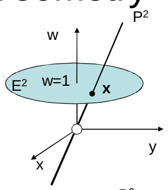
$\boldsymbol{P} = \boldsymbol{P}_0 + a_1(\boldsymbol{P}_1 - \boldsymbol{P}_0) \ldots + a_k(\boldsymbol{P}_k - \boldsymbol{P}_0)$

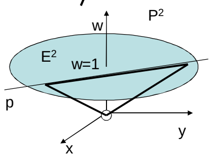
if $\boldsymbol{v}_i = \boldsymbol{P}_i - \boldsymbol{P}_0$ then

$\boldsymbol{P} = \boldsymbol{P}_0 + a_1\boldsymbol{v}_1 \ldots + a_k\boldsymbol{v}_k$

Vectors and Points in Geometry

- Projective extension of the affine space
- A point ***x*** in E^2 is defined with coordinates $\boldsymbol{X} = (X,Y)$ or as a point $\boldsymbol{x} = [x,y:w]^T$ with homogeneous coordinates $X = x / w$ $Y = y / w$, $w \neq 0$
- The point ***x*** in E^2 is a line (without an origin) in the projective space P^2
- A point ***x*** in E^3 is defined with coordinates $\boldsymbol{X} = (X,Y,Z)$ or as a point $\boldsymbol{x} = [x,y,z:w]^T$, with homogeneous coordinates $X = x / w$ $Y = y / w$, $Z = z / w$ $w \neq 0$
- A line in E^2 is a plane (without an origin) in the projective space P^2





Vectors and Points in Geometry

Vector representation $\boldsymbol{v} = (v_x, v_y, v_z : 0)$

Point representation $\boldsymbol{P} = (P_x, P_y, P_z : 1)$

Many libraries do not distinguish between points and vectors and treat them in the same manner

!! **BE CAREFUL** !!

Often used:

$\boldsymbol{v} = \boldsymbol{P}_1 - \boldsymbol{P}_0 = (P_{x1}, P_{y1}, P_{z1} : 1) - (P_{x0}, P_{y0}, P_{z0} : 1) = (P_{x1} - P_{x0}, P_{y1} - P_{y0}, P_{z1} - P_{z0} : 0) = (v_x, v_y, v_z : 0)$

!!! Do not make it on CPU/GPU – result $(v_x, v_y, v_z : \varepsilon)$ that is a point $(v_x / \varepsilon, v_y / \varepsilon, v_z / \varepsilon)$ in E^3

Vectors and Points in Geometry

How to handle vectors? [**X, V** - Euclidean , **x, v** – homogeneous]

$$\mathbf{V} = \mathbf{X}_1 - \mathbf{X}_0 = \left[\frac{x_1}{w_1} - \frac{x_0}{w_0}, \frac{y_1}{w_1} - \frac{y_0}{w_0}, \frac{z_1}{w_1} - \frac{z_0}{w_0}\right]^T$$

$$\mathbf{v} = [\mathbf{x}_1 : w_1]^T - [\mathbf{x}_0 : w_0]^T = [w_0\mathbf{x}_1 - w_1\mathbf{x}_0 : w_0 w_1]^T$$

- **Result** is a **vector** with homogeneous coordinate “hiding” the division operation
- **Vectors and points have “similar” representations, but different interpretation**
- if the vector length is not important – homogeneous coordinate value can be ignored [one division is saved]
- stability of computation –
- we do not need to solve instability

$$[\mathbf{x}_1 : \cong 1]^T - [\mathbf{x}_0 : \cong 1]^T = [\mathbf{x}' : \cong 0]$$

e.g. $(X,Y) = (v_x/\varepsilon, v_y/\varepsilon)$

where $\varepsilon \to 0$

Vectors and Points in Geometry

Projective extensions

$$\mathbf{A.B} = \left[\frac{a_x}{w_a}\frac{b_x}{w_b}, \frac{a_y}{w_a}\frac{b_y}{w_b}, \frac{a_z}{w_a}\frac{b_z}{w_b}\right] = \frac{1}{w_a w_b}[\mathbf{a} : w_a].[\mathbf{b} : w_b] = \frac{1}{w_a w_b}\mathbf{a.b}$$

$$\mathbf{A} \times \mathbf{B} = \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ A_x & A_y & A_z \\ B_x & B_y & B_y \end{bmatrix} = \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ \frac{a_x}{w_a} & \frac{a_y}{w_a} & \frac{a_z}{w_a} \\ \frac{b_x}{w_b} & \frac{b_y}{w_b} & \frac{b_z}{w_b} \end{bmatrix} = \frac{1}{w_a w_b}\begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ a_x & a_y & a_z \\ b_x & b_y & b_y \end{bmatrix} =$$

$$\frac{1}{w_a w_b}[\mathbf{a} : w_a] \times [\mathbf{b} : w_b] = \frac{1}{w_a w_b}\begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ a_x & a_y & a_z \\ b_x & b_y & b_y \end{bmatrix}$$

Vectors and Points in Geometry

Geometric product

$$\mathbf{AB} = \mathbf{A.B} + \mathbf{A} \wedge \mathbf{B} \qquad \mathbf{BA} = \mathbf{B.A} + \mathbf{B} \wedge \mathbf{A} = \mathbf{A.B} - \mathbf{A} \wedge \mathbf{B}$$

It can be seen

$$\mathbf{A.B} = \frac{1}{2}(\mathbf{AB} + \mathbf{BA}) \qquad \mathbf{A} \wedge \mathbf{B} = \frac{1}{2}(\mathbf{AB} - \mathbf{BA})$$

Result is a scalar and bivector in E^n

In the case of E^3 the operator $\wedge$ is analog to $\times$

Extension to the projective case is analogical

Generally: scalar, vector, bivector, trivector,.......

$\Rightarrow$ k-Blades

Vectors and Points in Geometry

$$\mathbf{A} = a_0 + a_1\mathbf{e}_1 + a_2\mathbf{e}_2 + a_3\mathbf{e}_{12}$$
$$\mathbf{B} = b_0 + b_1\mathbf{e}_1 + b_2\mathbf{e}_2 + b_3\mathbf{e}_{12}$$
$$\mathbf{A} + \mathbf{B} = (a_0 + a_1\mathbf{e}_1 + a_2\mathbf{e}_2 + a_3\mathbf{e}_{12}) + (b_0 + b_1\mathbf{e}_1 + b_2\mathbf{e}_2 + b_3\mathbf{e}_{12})$$
$$\mathbf{AB} = (a_0 + a_1\mathbf{e}_1 + a_2\mathbf{e}_2 + a_3\mathbf{e}_{12}).(b_0 + b_1\mathbf{e}_1 + b_2\mathbf{e}_2 + b_3\mathbf{e}_{12}) =$$
$$a_0b_0 + a_0b_1\mathbf{e}_1 + a_0b_2\mathbf{e}_2 + a_0b_3\mathbf{e}_3 + a_1b_0\mathbf{e}_1 + a_1b_1\mathbf{e}_1^2 + a_1b_2\mathbf{e}_{12} + ...$$
$$a_3b_2\mathbf{e}_{122} + a_3b_3\mathbf{e}_{12}^2$$

assumming $\mathbf{e}_1^2 = \mathbf{e}_1^2 = 1 \quad \mathbf{e}_{12} = \mathbf{e}_{21} \quad \mathbf{e}_{12}^2 = -1$

$$\mathbf{AB} = (a_0b_0 + a_1b_1 + a_2b_2 - a_3b_3) + (a_0b_1 + a_1b_0 + a_3b_2 - a_2b_3)\mathbf{e}_1 +$$
$$(a_0b_2 + a_1b_3 + a_2b_0 - a_3b_1)\mathbf{e}_2 + (a_0b_3 + a_1b_2 + a_3b_0 - a_2b_1)\mathbf{e}_{12}$$

Vectors and Points in Geometry

Geometric product

Blade	k	Sign
scalar	0	+
vector	1	+
bivector	2	-
trivector	3	-
4-vector	4	+

Vectors and Points in Geometry

Geometric product – “limited” projective extension

$$[\mathbf{a} : w_a][\mathbf{b} : w_b] = [\mathbf{a} : w_a].[\mathbf{b} : w_a] + [\mathbf{a} : w_a] \times [\mathbf{b} : w_a] =$$
$$\frac{1}{w_a w_b}\{\mathbf{a.b} + \mathbf{a} \times \mathbf{b}\} = [\mathbf{a.b} + \mathbf{a} \times \mathbf{b} : w_a w_b]$$

WOW !

No division operation is needed!

Vectors and Matrices

Vector – any vector in n-dimensional vector space ***V*** can be represented as

$$\boldsymbol{x} = x_1\boldsymbol{v}_1 + x_2\boldsymbol{v}_2 + \ldots + x_n\boldsymbol{v}_n$$

where $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n\}$ is a basis vector of ***V***

Notation

column $\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_{\ldots} \\ v_n \end{bmatrix}$ row $\mathbf{v} = \begin{bmatrix} v_1 & v_2 & v_{\ldots} & v_n \end{bmatrix}$

Vectors and Matrices

Matrix – square matrix n = m

$\boldsymbol{A}_{n,m} = [a_{ij}]$ $\qquad \mathbf{A} = \begin{bmatrix} a_{1,1} & \ldots & a_{1,m} \\ : & & : \\ a_{n,m} & \ldots & a_{n,m} \end{bmatrix}$

Special cases

m = 1 - column matrix – vector
(used in the right handed coordinate system)

n = 1 - row matrix - vector

Vectors and Matrices

Algebraic rules

$\boldsymbol{A} + \boldsymbol{B} = \boldsymbol{B} + \boldsymbol{A}$

$\boldsymbol{A} + (\boldsymbol{B} + \boldsymbol{C}) = (\boldsymbol{A} + \boldsymbol{B}) + \boldsymbol{C}$

$\boldsymbol{A} + \boldsymbol{0} = \boldsymbol{A}$

$\boldsymbol{A} + (-\boldsymbol{A}) = \boldsymbol{0}$

$a(\boldsymbol{A} + \boldsymbol{B}) = a\boldsymbol{A} + a\mathbf{B}$

$(a+b)\boldsymbol{A} = a\boldsymbol{A} + b\boldsymbol{A}$

$1 \,.\, \boldsymbol{A} = \boldsymbol{A}$

Transpose of a matrix $\boldsymbol{A}$ is a matrix $\boldsymbol{B} = \boldsymbol{A}^T$

where: $b_{ij} = a_{ij}$

Additional rules

$(\boldsymbol{A}^T)^T = \boldsymbol{A}$

$\boldsymbol{A}^{-1}\boldsymbol{A} = \boldsymbol{I}$

$(\boldsymbol{A}^{-1})^T = (\boldsymbol{A}^T)^{-1} = \boldsymbol{A}^{-T}$ (n x n)

$(a\boldsymbol{A})^T = a\boldsymbol{A}^T$

$(\boldsymbol{A}+\boldsymbol{B})^T = \boldsymbol{A}^T + \boldsymbol{B}^T$

Vectors and Matrices

Matrix multiplication

$\boldsymbol{C} = \boldsymbol{A}\,\boldsymbol{B}$

$$c_{ik} = \sum_{k=1}^{p} a_{ik} b_{kj}$$

$\boldsymbol{A} = [a_{ik}]$ i = 1,…,n, k = 1,…,p

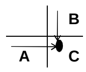
$\boldsymbol{B} = [b_{kj}]$ k = 1,…,p, j = 1,…,m

$\boldsymbol{C} = [c_{ij}]$ i = 1,…,n, j = 1,…,m

NOTE!

$\boldsymbol{A}\,\boldsymbol{B} \neq \boldsymbol{B}\,\boldsymbol{A}$

How to compute **C**?



Vectors and Matrices

$\boldsymbol{A}(\boldsymbol{BC}) = (\boldsymbol{AB})\boldsymbol{C}$ $\quad a(\boldsymbol{BC}) = (a\boldsymbol{B})\boldsymbol{C}$

$\boldsymbol{A}(\boldsymbol{B} + \boldsymbol{C}) = \boldsymbol{AB} + \boldsymbol{AC}$ $\quad (\boldsymbol{A} + \boldsymbol{B})\boldsymbol{C} = \boldsymbol{AC} + \boldsymbol{BC}$

$(\boldsymbol{AB})^T = \boldsymbol{B}^T\boldsymbol{A}^T$

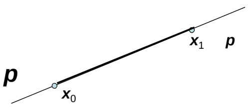
Block matrices

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right]\left[\begin{array}{c|c} \mathbf{E} & \mathbf{F} \\ \hline \mathbf{G} & \mathbf{H} \end{array}\right] = \left[\begin{array}{c|c} \mathbf{AE} + \mathbf{BG} & \mathbf{AF} + \mathbf{BH} \\ \hline \mathbf{CE} + \mathbf{DG} & \mathbf{CF} + \mathbf{DH} \end{array}\right]$$

Lines and Line Segments

Lines in E^2

two points define a line ***p***



- implicit description

$$a x + b y + d = 0$$
$$\mathbf{a}^T \mathbf{x} + d = 0$$

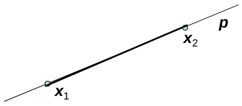
- parametric description

$$\mathbf{x}(t) = \mathbf{x}_0 + (\mathbf{x}_1 - \mathbf{x}_0)t$$
$$t \in (-\infty, \infty), \quad \text{resp. } t \in \langle 0, 1 \rangle$$

Lines and Line Segments

Lines in E^2

- two points define a line ***p***
- 2 equations for 3 parameters – a, b, d
- linear homogeneous system, i.e. one parametric solution



$$a x_1 + b y_1 + d = 0$$
$$a x_2 + b y_2 + d = 0$$

$$\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{bmatrix}\begin{bmatrix} a \\ b \\ d \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Lines and Line Segments

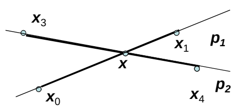
A point $\boldsymbol{x}$ as an intersection of two lines in E^2

- system of linear equations must be solved
- numerical stability of a solution - intersection x collinearity

$$a_1 x + b_1 y + d_1 = 0$$
$$a_2 x + b_2 y + d_2 = 0$$

$$\begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -d_1 \\ -d_2 \end{bmatrix}$$

$$\mathbf{A}\,\mathbf{x} = \mathbf{b}$$



Points and Planes in E³

- A point ***x*** as an intersection of three planes in E³
- system of linear equations must be solved
- numerical stability of the solution
  if |det A| < ε then "singular case"??
  What is ε value ??

$$a_1 x + b_1 y + c_1 z + d_1 = 0$$
$$a_2 x + b_2 y + c_2 z + d_2 = 0$$
$$a_3 x + b_3 y + c_3 z + d_3 = 0$$

$$\begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -d_1 \\ -d_2 \\ -d_3 \end{bmatrix}$$
$$\mathbf{A}\,\mathbf{x} = \mathbf{b}$$

Points and Planes in E³

- Three points define a plane ρ
- Homogeneous system must be solved
- Non unique solution

$$a x + b y + c z + d = 0$$
$$\mathbf{a}^T \mathbf{x} + d = 0$$

$$ax_1 + ay_1 + az_1 + d = 0$$
$$ax_2 + ay_2 + az_2 + d = 0$$
$$ax_3 + ay_3 + az_3 + d = 0$$

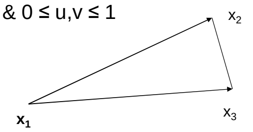
$$\begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$
$$\mathbf{A}\mathbf{x} = \mathbf{0}$$

Points and Planes in E³

- A parametric form for the plane ρ

$$\mathbf{x}(u,v) = \mathbf{x}_1 + (\mathbf{x}_2 - \mathbf{x}_1)u + (\mathbf{x}_3 - \mathbf{x}_1)v$$
$$u, v \in (-\infty, \infty)$$

Triangle: 0 ≤ |u+v| ≤ 1 & 0 ≤ u,v ≤ 1



Interpolation

- Linear interpolation - formulation
  - Parametric
  - Implicit form
- Quadric/cubic interpolation – splines etc. [not discussed here]
- Radial basis functions RBF - **multidimensional**

$$f(\mathbf{x}) = \sum_{i=1}^{N} \lambda_i f(\|\mathbf{x} - \mathbf{x}_i\|) + p(\mathbf{x}_i)$$
$$p(\mathbf{x}) = \mathbf{a}^T \mathbf{x} + d \qquad \sum \lambda_i = 0 \qquad \sum \lambda_i \mathbf{x}_i = \mathbf{0}$$

Interpolation

$$\begin{bmatrix} f_{11} & \cdots & f_{1n} & x_1 & y_1 & z_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ f_{n1} & \cdots & f_{nn} & x_n & y_n & z_n & 1 \\ x_1 & \cdots & x_n & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ z_1 & \cdots & z_n & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \\ a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} h_1 \\ \vdots \\ h_n \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where: $f_{ij} = f(\|\mathbf{x}_i - \mathbf{x}_j\|)$ $\quad r_i = \|\mathbf{x}_i - \mathbf{x}\|$

$f(r) = r^2 \lg(r)$ *etc.*

RBF Interpolation

Corrupted image 30%

Reconstructed image

RBF Interpolation

Corrupted image 90%   Reconstructed image

Principle of Duality

- Any theorems remain true in E^2 when we interchange words “point” and “line”, “lie on” and “pass through”, “join” and “intersection” etc.
- Points and planes are dual in E^3 etc. (not points and lines!)
- It means that intersection computation of two lines and a line given by two points should be same if we use the principle of duality.

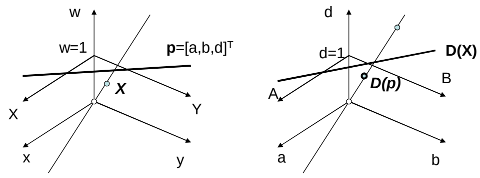
Principle of Duality

- In E^2
  - a line **p** is given by two points (how to compute 3 coefficients $[a,b,d]^T$ of the line ax+by+d=0 ?)
  - a point **x** as an intersection of two lines
- In E^3
  - a plane **ρ** is given by three points (how to compute 4 coefficients $[a,b,c,d]^T$ of the plane ax+by+cz+d=0 ?)
  - a point **x** as an intersection of three planes

Dual problems – but computations are not “symmetrical”
Stability and robustness??

WHAT IS WRONG ??

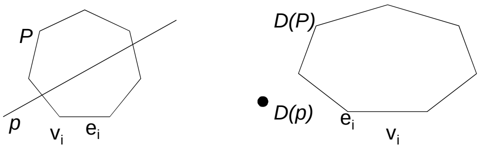
Principle of Duality



$ax + by + d = 0$

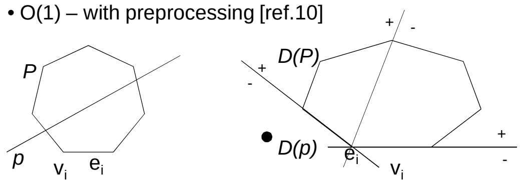
Duality in Geometric Algorithms

Test if a line *p* intersects a convex polygon *P*
is dual to a test if a point ***x***=*D(**p**)*
is outside of a convex polygon P’=D(P)



Duality in Geometric Algorithms

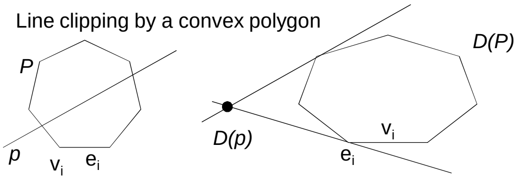
Test Point-in-Polygon is of complexity:
- O(N) – test again half-spaces
- O(lg N) – test using binary search over indexes
- O(1) – with preprocessing [ref.10]



The same complexity for a test
if a line intersects a convex polygon !!!

Duality in Geometric Algorithms

Line clipping by a convex polygon



We look after "touching" lines using LOG search
=> **There is O(lg N) line clipping algorithm for E2** , Ref.[11]

We used the Duality principle to derive a NEW algorithm !!

Computation in Projective Space

Intersection point ***x*** given by two lines $\mathbf{p}_1=[a_1,b_1,d_1]^T$ and $\mathbf{p}_2=[a_2,b_2,d_2]^T$

$$a_1 X + b_1 Y + d_1 = 0$$
$$a_2 X + b_2 Y + d_2 = 0$$
$$\mathbf{A}\,\mathbf{X} = -\mathbf{d} \qquad /*\ w \neq 0$$

$\boldsymbol{x} = \mathbf{p}_1 \times \mathbf{p}_2$

$$a_1 x + b_1 y + d_1 w = 0$$
$$a_2 x + b_2 y + d_2 w = 0$$
$$\mathbf{B}\,\mathbf{x} = \mathbf{0} \qquad \mathbf{x} = [x, y : w]^T$$

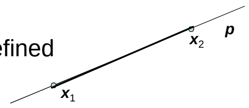
$$\mathbf{p}_1 \times \mathbf{p}_2 = \det \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ a_1 & b_1 & d_1 \\ a_2 & b_2 & d_2 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} a_1 & b_1 & d_1 \\ a_2 & b_2 & d_2 \end{bmatrix}$$

Computation in Projective Space

Lines in E^2

- a line $\boldsymbol{p}$ = [a,b,d] ^{T} is defined by points
  $\boldsymbol{x}_1=[x_1, y_1, w_1]^T$
  $\boldsymbol{x}_2=[x_2, y_2, w_2]^T$

  $\boldsymbol{p} = \boldsymbol{x}_1 \times \boldsymbol{x}_2$



$$\mathbf{x}_1 \times \mathbf{x}_2 = \det \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ x_1 & y_1 & w_1 \\ x_2 & y_2 & w_2 \end{bmatrix}$$

- we do not need to solve linear equations and no division operation is needed!
- stability evaluation AFTER computation

Computation in Projective Space

$$\mathbf{x}_1 \times \mathbf{x}_2 = \det \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ x_1 & y_1 & w_1 \\ x_2 & y_2 & w_2 \end{bmatrix} \qquad \mathbf{a} = [a \quad b \quad d]^T$$

It means

$$\mathbf{a}^T(\mathbf{x}_1 \times \mathbf{x}_2) = 0 \qquad \det \begin{bmatrix} a & b & d \\ x_1 & y_1 & w_1 \\ x_2 & y_2 & w_2 \end{bmatrix} = 0$$

Computation in Projective Space

In E^2

- A line $\boldsymbol{p}$ can be determined by the cross product of two points $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$
- An intersection $\boldsymbol{x}$ point can be determined by the cross product of two lines $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$
- If $\boldsymbol{x_1}$ or $\boldsymbol{x_2}$ are in homogeneous coordinates, e.g. $w_i \neq 1$ no division is needed

How the E^3 case is handled?

- A point is dual to a plane $\boldsymbol{\rho}$
  $\boldsymbol{\rho}=[a,b,c,d]^T$
- There is no "direct duality" for a line in E^3

Computation in Projective Space

- Cross product definition
- A plane $\boldsymbol{\rho}$ is determined as a cross product of three given points

$$\mathbf{x}_1 \times \mathbf{x}_2 \times \mathbf{x}_3 = \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} & \mathbf{l} \\ x_1 & y_1 & z_1 & w_1 \\ x_2 & y_2 & z_2 & w_2 \\ x_3 & y_3 & z_3 & w_3 \end{bmatrix}$$

$$ax + by + cz + dw = 0 \qquad \mathbf{a}^T\mathbf{x} = 0$$

Due to the duality

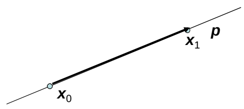
- An intersection point $\boldsymbol{x}$ of three planes is determined as a cross product of three given planes

$$\boldsymbol{\rho}_1 \times \boldsymbol{\rho}_2 \times \boldsymbol{\rho}_3 = \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} & \mathbf{l} \\ a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \\ a_3 & b_3 & c_3 & d_3 \end{bmatrix}$$

Computation in Projective Space

- No division operation!
- An “intersection of parallel lines” can be computed - it leads to $[x,y:0]^T$, resp. $[x,y:\varepsilon]^T$ – a point in, resp.close to infinity
- More robust computations in general
  no IF clauses (conditions) are needed
- Substantial speed-up on CPU or GPU can be expected due to vector-vector operations support

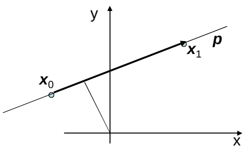
Computation in Projective Space



$$\mathbf{x}(t) = \mathbf{x}_0 + (\mathbf{x}_1 - \mathbf{x}_0)t$$
$$t \in (-\infty, \infty)$$

A line in E^3
- parametric form
- as an intersection of two planes
- Plücker coordinates

Computation in Projective Space

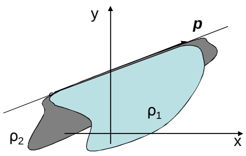


Plücker coordinates

$$\mathbf{p} = \left[(\mathbf{x}_1 - \mathbf{x}_2)^T : (\mathbf{x}_1 \times \mathbf{x}_2)^T\right]^T$$
$$\left[\boldsymbol{\omega}^T : \mathbf{v}^T\right]^T \quad \text{sometimes as } [\boldsymbol{\omega} : \mathbf{v}]^T$$

$$\mathbf{x}_i = [x_i, y_i, z_i, w_i]^T \quad i = 1,2$$
$$\mathbf{L} = \mathbf{x}_0\mathbf{x}_1^T - \mathbf{x}_1\mathbf{x}_0^T \quad \textit{tensor product} - \textit{matrix}$$
$$\boldsymbol{\omega} = [l_{41} \quad l_{42} \quad l_{43}]^T \qquad \mathbf{v} = [l_{23} \quad l_{31} \quad l_{12}]^T$$
$$\mathbf{q}(\tau) = \frac{\mathbf{v} \times \boldsymbol{\omega}}{\|\boldsymbol{\omega}\|^2} + \boldsymbol{\omega}\tau \qquad \mathbf{q}(\tau) = \begin{bmatrix} \mathbf{v} \times \boldsymbol{\omega} + \boldsymbol{\omega}\tau\|\boldsymbol{\omega}\|^2 \\ \|\boldsymbol{\omega}\|^2 \end{bmatrix}$$

Computation in Projective Space



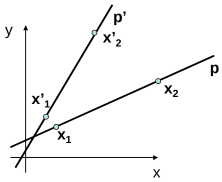
Plücker coordinates
- a line given by two planes – due to **DUALITY**

$$\mathbf{q}(\tau) = \frac{\boldsymbol{\omega} \times \mathbf{v}}{\|\mathbf{v}\|^2} + \mathbf{v}\tau$$
$$\boldsymbol{\omega} = [l_{41} \quad l_{42} \quad l_{43}]^T \qquad \mathbf{v} = [l_{23} \quad l_{31} \quad l_{12}]^T$$
$$\mathbf{L} = \mathbf{a}_0\mathbf{a}_1^T - \mathbf{a}_1\mathbf{a}_0^T \quad \textit{tensor product} - \textit{matrix}$$
$$\mathbf{a}_i = [a_i, b_i, c_i, d_i]^T \quad i = 1,2$$

Computation in Projective Space

- Geometric transformations
  – points

$$\mathbf{p} = \mathbf{x}_1 \times \mathbf{x}_2 = \det\begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ x_1 & y_1 & w_1 \\ x_2 & y_2 & w_2 \end{bmatrix}$$



$$\mathbf{x}' = \mathbf{Q}\,\mathbf{x}$$

a line given by two tranformed points ? $\mathbf{p}' = ?$

$$(\mathbf{Q}\,\mathbf{x}_1) \times (\mathbf{Q}\,\mathbf{x}_2) = \left(\mathbf{Q}^{-1}\right)^T \mathbf{x}_1 \times \mathbf{x}_2 \frac{1}{\det(\mathbf{Q})}$$

as $\mathbf{p}'$: $ax + by + cw = 0$

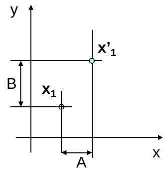
we can multiply Eq. by $\frac{1}{\det(\mathbf{Q})}$

$$\mathbf{p}' = \left(\mathbf{Q}^{-1}\right)^T \mathbf{x}_1 \times \mathbf{x}_2$$

Computation in Projective Space

- Translation
  – distance given in E^2
  – distance given in P^2

$$\mathbf{x}' = \begin{bmatrix} 1 & 0 & A \\ 0 & 1 & B \\ 0 & 0 & 1 \end{bmatrix}\mathbf{x}$$



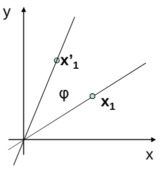
$$\mathbf{x}' = \begin{bmatrix} w & 0 & a \\ 0 & w & b \\ 0 & 0 & w \end{bmatrix}\mathbf{x} \quad (A, B) = \left(\frac{a}{w}, \frac{b}{w}\right) \quad w \neq 0$$

if A or B are fractions, we can avoid division !

Computation in Projective Space

- Rotation

$$\mathbf{x}' = \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}$$

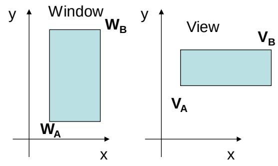


$$\mathbf{x}' = \begin{bmatrix} A & -B & 0 \\ B & A & 0 \\ 0 & 0 & C \end{bmatrix} \mathbf{x}$$

$$\cos\varphi = \frac{A}{C} \quad \sin\varphi = \frac{B}{C}$$

Computation in Projective Space

- Window-Viewport



$$\mathbf{T}_1 = \begin{bmatrix} 1 & 0 & -Wx_A \\ 0 & 1 & -Wy_A \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{T}_2 = \begin{bmatrix} 1 & 0 & -Vx_A \\ 0 & 1 & -Vy_A \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{S} = \begin{bmatrix} \Delta Vx/\Delta Wx & 0 & 0 \\ 0 & \Delta Vy/\Delta Wy & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{Q} = \mathbf{T}_2\ \mathbf{S}\ \mathbf{T}_1$$

$$\mathbf{x}' = \mathbf{Q}\ \mathbf{x}$$

$$\mathbf{S}' = \begin{bmatrix} \Delta Vx\Delta Wy & 0 & 0 \\ 0 & \Delta Vy\Delta Wx & 0 \\ 0 & 0 & \Delta Wx\Delta Wy \end{bmatrix} \quad \mathbf{Q}' = \mathbf{T}_2\ \mathbf{S}'\ \mathbf{T}_1$$

No division operation needed !!

Computation in Projective Space

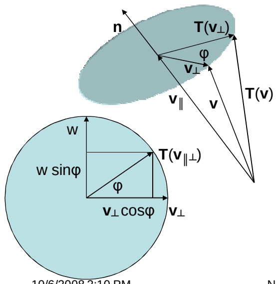
Fundamental geometric transformations

- translation, rotation, reflection, shearing, scaling, projection from E^3 to E^2
- can be performed without division operation and if parameters are given as fractions, matrices can be easily modified for parameters given in homogeneous representation

No division operation needed !!

Computation in Projective Space

General axis rotation



$\mathbf{T}(\mathbf{v}_\perp) = \mathbf{v}_\parallel \cos\varphi + \mathbf{w} \sin\varphi$
$\mathbf{T}(\mathbf{v}) = \mathbf{T}(\mathbf{v}_\parallel) + \mathbf{T}(\mathbf{v}_\perp)$

$$\mathbf{T} = \mathbf{I}\cos\varphi + (1-\cos\varphi)(\mathbf{n}\otimes\mathbf{n}) + (\mathbf{n}\times\mathbf{v})\sin\varphi \qquad \|\mathbf{n}\| = 1$$

$$\mathbf{n}\otimes\mathbf{n} = \mathbf{n}\mathbf{n}^T \quad \text{tensor product-matrix}$$

Interpolation

Linear parametrization

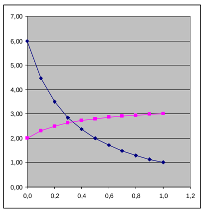
$$\mathbf{X}(t) = \mathbf{X}_0 + (\mathbf{X}_1 - \mathbf{X}_0)t \quad t \in (-\infty, \infty)$$

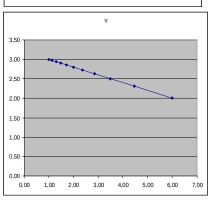
Non-linear (monotonous) parametrization

$$\mathbf{x}(t) = \mathbf{x}_0 + (\mathbf{x}_1 - \mathbf{x}_0)t \quad t \in (-\infty, \infty)$$

$$x(t) = x_0 + (x_1 - x_0)t \qquad y(t) = y_0 + (y_1 - y_0)t$$

$$z(t) = z_0 + (z_1 - z_0)t \qquad w(t) = w_0 + (w_1 - w_0)t$$





- It means that we can interpolate using homogeneous coordinates without a need of "normalization" to E^k !!
- Homogeneous coordinate w ≥ 0
- In many algorithms, we need "monotonous" parameterization, only !

Interpolation

Linear parametrization

$$\mathbf{X}(t) = \mathbf{X}_0 + (\mathbf{X}_1 - \mathbf{X}_0)u + (\mathbf{X}_2 - \mathbf{X}_0)v \qquad u, v \in (-\infty, \infty)$$

Non-linear (monotonous) parametrization

$$\mathbf{x}(t) = \mathbf{x}_0 + (\mathbf{x}_1 - \mathbf{x}_0)u + (\mathbf{x}_2 - \mathbf{x}_0)u \qquad u, v \in (-\infty, \infty)$$

$$x(t) = x_0 + (x_1 - x_0)u + (x_2 - x_0)v$$

$$y(t) = y_0 + (y_1 - y_0)u + (y_2 - y_0)v$$

$$z(t) = z_0 + (z_1 - z_0)u + (z_2 - z_0)v$$

$$w(t) = w_0 + (w_1 - w_0)u + (w_2 - w_0)v$$

- Homogeneous coordinate w ≥ 0
- In many algorithms, we need "monotonous" parameterization, only !

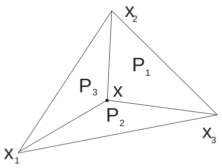
Barycentric coordinates

Let us consider a triangle with vertices $\mathbf{X}_1$, $\mathbf{X}_2$, $\mathbf{X}_3$,
A position of any point $\mathbf{X} \in E^2$ can be expressed as

$$a_1 X_1 + a_2 X_2 + a_3 X_3 = X$$
$$a_1 Y_1 + a_2 Y_2 + a_3 Y_3 = Y$$

additional condition

$$a_1 + a_2 + a_3 = 1 \quad 0 \le a_i \le 1$$
$$a_i = \frac{P_i}{P} \quad \mathrm{i} = 1,...,3$$



Linear system must be solved
If points $\mathbf{x}_i$ are given as $[x_i, y_i, z_i: w_i]^T$ and $w_i \ge 0$
then $\mathbf{x}_i$ must be "normalized"

Barycentric coordinates

It can be modified to:

$$b_1 X_1 + b_2 X_2 + b_3 X_3 + b_4 X = 0$$
$$b_1 Y_1 + b_2 Y_2 + b_3 Y_3 + b_4 Y = 0$$
$$b_1 + b_2 + b_3 + b_4 = 0$$
$$b_i = -a_i b_4 \quad i = 1,...,3 \quad b_4 \ne 0$$

Rewriting

$$\begin{bmatrix} X_1 & X_2 & X_3 & X \\ Y_1 & Y_2 & Y_3 & Y \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \mathbf{0}$$

$$\mathbf{b} = \boldsymbol{\xi} \times \boldsymbol{\eta} \times \mathbf{w}$$
$$\mathbf{b} = [b_1, b_2, b_3, b_4]^T$$
$$\boldsymbol{\xi} = [X_1, X_2, X_3, X]^T$$
$$\boldsymbol{\eta} = [Y_1, Y_2, Y_3, Y]^T$$
$$\mathbf{w} = [1,1,1,1]^T$$

Barycentric coordinates

if $w_i \ge 0$

$$\begin{bmatrix} x_1 & x_2 & x_3 & x \\ y_1 & y_2 & y_3 & y \\ w_1 & w_2 & w_3 & w \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \mathbf{0}$$

$$\mathbf{b} = \boldsymbol{\xi} \times \boldsymbol{\eta} \times \mathbf{w}$$
$$\mathbf{b} = [b_1, b_2, b_3, b_4]^T$$
$$\boldsymbol{\xi} = [x_1, x_2, x_3, x]^T$$
$$\boldsymbol{\eta} = [y_1, y_2, y_3, y]^T$$
$$\mathbf{w} = [w_1, w_2, w_3, w]^T$$

Barycentric coordinates

if $w_i \ge 0$

$$\mathbf{b} = \boldsymbol{\xi} \times \boldsymbol{\eta} \times \mathbf{w}$$
$$\mathbf{b} = [b_1, b_2, b_3, b_4]^T$$
$$\boldsymbol{\xi} = [x_1, x_2, x_3, x]^T$$
$$\boldsymbol{\eta} = [y_1, y_2, y_3, y]^T$$
$$\mathbf{w} = [w_1, w_2, w_3, w]^T$$

$$0 \le (-b_1 : w_2 w_3 w) \le 1$$
$$0 \le (-b_2 : w_3 w_1 w) \le 1$$
$$0 \le (-b_3 : w_1 w_2 w) \le 1$$

It means that we can compute barycentric coordinates without division operation

Simple modification for the position in a tetrahedron [4]

Length, Area and Volume

Length, area and volume computation in projective space
if an element is given by points in homogeneous coordinates

Line segment length

$$\mathbf{p} = \mathbf{x}_1 \times \mathbf{x}_2 \qquad \mathbf{p} = [a, b, d]^T \quad \mathbf{n} = [a, b]^T$$
$$l = \left| \left(\sqrt{\mathbf{n}^T \mathbf{n}} : w_1 w_2 \right) \right|$$

Triangle area

$$\boldsymbol{\rho} = \mathbf{x}_1 \times \mathbf{x}_2 \times \mathbf{x}_3 \qquad \boldsymbol{\rho} = [a, b, c, d]^T \quad \mathbf{n} = [a, b, c]^T$$
$$S = \left| \left(\sqrt{\mathbf{n}^T \mathbf{n}} : 2 w_1 w_2 w_3 \right) \right|$$

Length, Area and Volume

Tetrahedron volume

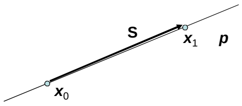
$$\boldsymbol{\rho} = \mathbf{x}_1 \times \mathbf{x}_2 \times \mathbf{x}_3 \times \mathbf{x}_4 \qquad \boldsymbol{\rho} = [a, b, c, d, e]^T \quad \mathbf{n} = [a, b, c, d]^T$$
$$V = \left| \left(\sqrt{\mathbf{n}^T \mathbf{n}} : 6 w_1 w_2 w_3 w_4 \right) \right|$$

General formula

$$Q_k = \left| \left(\sqrt{\mathbf{n}^T \mathbf{n}} : (k-1)! \prod_{i=1}^{k} w_i \right) \right| \quad \text{k = number of end-points}$$

$$\bar{\mathbf{n}} = \frac{\mathbf{n}}{\sqrt{\mathbf{n}^T \mathbf{n}}} = \left(\mathbf{n} : (k-1)! \, Q_k \prod_{i=1}^{k} w_i \right)$$

Intersection Computation in Projective Space



$$\mathbf{X}(t) = \mathbf{X}_0 + (\mathbf{X}_1 - \mathbf{X}_0)t \quad t \in (-\infty, \infty)$$
$$\mathbf{A}^T\mathbf{X} + d = 0 \qquad ax + by + cz + d = 0$$
$$\mathbf{A} = [a, b, c]^T \qquad \mathbf{S} = \mathbf{X}_1 - \mathbf{X}_0$$
$$t = -\frac{\mathbf{A}^T\mathbf{X}_0 + d}{\mathbf{A}^T\mathbf{S}}$$

- Linear interpolation& parameterization very often used
- Intersection of a line and a plane
- Robustness problems if $\mathbf{A}^T\mathbf{S} \to 0$

??? How to avoid an instability ???

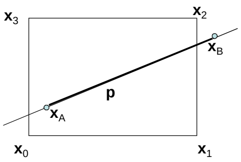
Intersection Computation in Projective Space

$$\mathbf{x}(t) = \mathbf{x}_0 + (\mathbf{x}_1 - \mathbf{x}_0)t \quad t \in (-\infty, \infty)$$
$$\mathbf{a}^T\mathbf{x} = 0 \qquad ax + by + cz + d = 0$$
$$\mathbf{a} = [a, b, c, d]^T \qquad \mathbf{S} = \mathbf{X}_1 - \mathbf{X}_0$$
$$t = -\frac{\mathbf{a}^T\mathbf{x}_0}{\mathbf{a}^T\mathbf{s}}$$
$$\tau = -\mathbf{a}^T\mathbf{x}_0 \qquad \tau_w = \mathbf{a}^T\mathbf{s}$$
$$\mathbf{t} = [\tau : \tau_w] \quad \textbf{if } \tau_w \le 0 \textbf{ then } \mathbf{t} := -\mathbf{t}$$
TEST
$$\textbf{if } t > t_{\min} \textbf{ then.....}$$
$$\textbf{if } \tau * \tau_{\min_w} > \tau_w * \tau_{\min} \textbf{ then.....} \quad \text{condition } \tau_w \ge 0$$

- An intersection of a plane with a line in E^2 can be computed efficiently [6]
- Comparison operations must be modified !!!
- Cyrus-Beck line clipping algorithm 10-25% faster

Intersection Computation in Projective Space

Line clipping in E^2 algorithms
- Cohen-Sutherland
- Liang-Barsky
- Hodgman
- Skala – modification of Clip_L for line segments



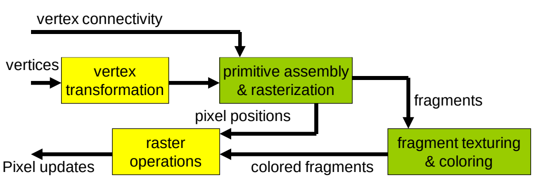
procedure CLIP_L; {details in [3]}
{ $\mathbf{x}_A$, $\mathbf{x}_B$ – in homogeneous coordinates }
{ The **EXIT** ends the procedure }
{ **input**: $\mathbf{x}_A$, $\mathbf{x}_B$; $\mathbf{x}_A = [x_A, y_A, 1]^T$ $\mathbf{p} = [a,b,c]^T$ }
begin
{1} $\mathbf{p} := \mathbf{x}_A \times \mathbf{x}_B$; { ax+by+c = 0}
{2} **for** k:=0 **to** N-1 **do** { $\mathbf{x}_k = [x_k, y_k, 1]^T$ }
{3} **if** $\mathbf{p}^T\mathbf{x}_k \ge 0$ **then** $\mathbf{c}_k$:=1
else c_k:=0;
{4} **if** **c** = $[0000]^T$ **or** **c** = $[1111]^T$
then EXIT;
{5} i:= TAB1[**c**]; j:= TAB2[**c**];
{6} $\mathbf{x}_A := \mathbf{p} \times \mathbf{e}_i$; $\mathbf{x}_B := \mathbf{p} \times \mathbf{e}_j$;
{7} **DRAW** ($\mathbf{x}_A$; $\mathbf{x}_B$) {$\mathbf{e}_i$ – *i-th edge* }
end {CLIP_L};

Intersection Computation in Projective Space

Iterative computations
- values are represented as fractions with floats
- exponents grow – need of “exponents normalization”
  - not available on current CPUs
  - necessity of explicit CALL
- solution - see PLib for .NET [8]

Cg / HLSL and GPU Computing

- GPU (Graphical Processing Unit) -optimized for matrix-vector, vector-vector operation – especially for [x,y,z:w]
- Native arithmetic operations with homogeneous coordinates – without exponent “normalization”
- Programmable HW – parallel processing



Cg / HLSL and GPU Computing

- 4D cross product can be implemented in Cg/HLSL on GPU (not optimal implementation) as:

```
float4 cross_4D(float4 x1, float4 x2, float4 x3)
{ float4 a;
   a.x=dot(x1.yzw, cross(x2.yzw, x3.yzw));
   a.y=-dot(x1.xzw, cross(x2.xzw, x3.xzw));
   a.z=dot(x1.xyw, cross(x2.xyw, x3.xyw));
   a.w=-dot(x1.xyz, cross(x2.xyz, x3.xyz));
   return a;
}
```

Conclusion

- Fundamentals of computation in the projective space have been introduced
- Proposed approach helps to improve robustness of algorithms, but it does not give the ultimate solution - limited numerical precision
- Homogeneous coordinate *w* must be non-negative (simplification comparison operations)
- Comparison operations are a little bit complicated – but tests rely on separation functions – higher robustness
- Due to GPU and CPU architecture algorithms might be significantly faster even in SW implementation

Conclusion

- A new data type for programming languages – float_projective, double_projective should be considered
  - perhaps as a native representation
  - it enables more robust numerical algorithms
  - unfortunately increases a data bus traffic
  - operation "exponent normalization" should be supported on CPU/GPU in HW – significantly slow in SW
  - experimental library PLib is available [8]
- Geometry algebra applications in CG & CV ??

References

[1] van Verth,J.M., Bishop,L.M.: Essential Mathematics for Games and Interactive Applications, Morgan Kaufmann 2005
[2] Skala,V.: GPU Computation in Projective Space Using Homogeneous Coordinates , Game Programming GEMS 6 (Ed.Dickheiser,M.), pp.137-147, Charles River Media, 2006
[3] Skala,V.: A new approach to line and line segment clipping in homogeneous coordinates, The Visual Computer, Vol.21, No.11, pp.905-914, Springer , 2005
[4] Skala,V.: Barycentric coordinates computation in homogeneous coordinates, Computers&Graphics, Elsevier, pp.120-127, 2007
[5] Skala,V.: Length, Area and Volume Computation in Homogeneous Coordinates, International Journal of Image and Graphics, Vol.6., No.4, pp.625-639, 2006
[6] Yamaguchi,F.: Computer Aided Design – A totally Four-Dimensional Approach, Springer Verlag 2002
[7] Fernando,R., Kilgard,M.J.: The Cg Tutorial, Addison Wesley, 2003
[8] Skala,V.: Intersection Computation in Projective Space using Homogeneous coordinates, accepted for publication, International Journal of Image and Graphics, 2008
[9] Skala,V., Kaiser,J., Ondracka,V.: Library for Computation in the Projective Space, Aplimat 2007 conf., 2007
[10] Skala,V.: Line Clipping in E2 with Suboptimal Complexity O(1), Computers & Graphics, Vol.20, No.4, pp.523-530, 1996
[11] Skala,V.: O(lg N) Line Clipping Algorithm in E2, Computers & Graphics, Pergamon Press, Vol.18, No.4, 1994.

References

Additional readings:

- Dorst,L., Fontijne,D., Mann,S.: Geometric Algebra for Computer Science - An Object-oriented Approach to Geometry, Elsevier, 2007
- Nielsen,F: Visual Computing: Geometry, Graphics and Vision, Charles River Media, 2005
- Hartley,R., Zisserman,A.: Multiple View Geometry in Computer Vision, Cambridge Univ.Press, 2000
- Farin,G., Hansford,D.: Practical Linear Algebra – A Geometry Toolbox, A.K.Peters, 2005
- Doran,Ch., Lasenby,A.: Geometric Algebra for Physicist, Cambridge Univ.Press, 2005
- Hanák,I.: The GPU and Graphic Algorithms http://www.kiv.zcu.cz/publications/2005/tr-2005-05.pdf

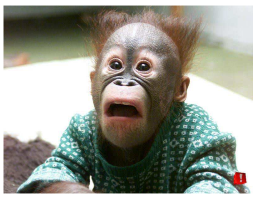
Acknowledgment

Thanks belong to many colleagues and students at the University of West Bohemia for their critical comments and suggestions.

Activities supported by:
- INTUITION, project FP6 NoE
- VIRTUAL, project MSMT Czech Rep.,No.2C06002
- LC CPG, project MSMT Czech Rep., No.LC06008

Invitation WSCG2009 – 17th Int.Conf.in Central Europe on Computer Graphics, Visualization and Computer Vision (VR,AR, Haptics,..covered) **http://wscg.zcu.cz - supported by INTUITION**

Thank you for your attention



Questions ??

Contact
Vaclav Skala
skala@kiv.zcu.cz
subj. Intuition 2008 - Tutorial

Center of Computer Graphics and Visualization http://herakles.zcu.cz
Department of Informatics and Computer http://www.kiv.zcu.cz
Science Faculty of Applied Sciences http://www.fav.zcu.cz
University of West Bohemia, Univerzitni 8 http://www.zcu.cz
WSCG conferences [since 1992] http://wscg.zcu.cz
CZ 306 14 Plzen
Czech Republic

INTUITION 2008 Programme – Tutorials (Oct 6) and Conference Day 1 (Oct 7)

Monday October 6, 2008

14:00 - 19:30 — **Parallel Tutorial Session:**

Mathematical Fundamentals for Computer Graphics and Virtual Reality
Prof. Vaclav Skala - University of West Bohemia, Czech Republic

Building a Complete Virtual Reality Application
Franco Tecchia, Marcello Carrozzino - PERCRO / Scuola Superiore S.Anna, Italy

Tuesday October 7, 2008

08:30 - 09:30 — Registration

09:30 - 09:40 — **INTUITION Official Opening** by the Organiser and the INTUITION Coordinator
Giuseppe Varalda - CRF, Angelos Amditis – ICCS

09:40 - 10:00 — **Conference Overview** by the IPC Chairs
Massimo Bergamasco - PERCRO, Roland Blach – FhG-IAO, Giannis Karaseitanidis – ICCS

10:00 - 10:10 — **Welcome address by the Turin Chamber of Commerce**
Mr. Alessandro Barberis, President

10:10 - 10:20 — **Welcome address by ATA**
Mr. Nevio di Giusto, President of ATA and CEO of FIAT Research Centre and ELASIS

10:20 - 10:30 — **Welcome address by MIMOS**
Mr. Davide Borra, President of MIMOS and President of NoReal

10:30 - 11:15 — **Keynote Speech 1: Game-Based Virtual Rehabilitation**
Dr. Grigore Burdea – Rutgers University, USA

11:15 - 11:30 — *Coffee Break*

11:30 - 12:15 — **Keynote Speech 2: Virtual Environments for Motor Skills Acquisition and Transfer**
Dr. Massimo Bergamasco – Scuola Superiore S.Anna, Italy

12:15 - 12:45 — **EuroVR Association**
Angelos Amditis – ICCS

12:45 - 13:45 — *Lunch*

13:45 - 15:30 — **Parallel Session I: Haptics – *Chair: Jerome Perret (HAPTION, FR)***

Force-Feedback, from Teleoperation to Simulation in Virtual Reality [1]
Jérôme Perret, Loïc Tching

Haptic Rendering of Highly Complex Quasi-Convex Objects
Huagen Wan, Xiaoxia Han, Zhihua Zhou

The Design of a Haptic Simulator for Teaching and Assessing Spinal Anaesthesia
Erik Lövquist, Zsuzsanna Kulcsár, George D. Shorten, Annette Aboulafia, Mikael Fernström

Hardware-Based Parallel Computing for Real-Time Haptic Rendering of Deformable Objects
Ramin Mafi, Shahin Sirouspour, Behzad Mahdavikhah, Brian Moody, Kaveh Elizeh

13:45 - 15:30 — **Parallel Session II: Applications I [2] – *Chair: Cristiano Montrucchio (Alenia Aeronautica, IT)***

COSE Centre: The Thales Alenia Space Italia Collaborative System Engineering Centre
Valter Basso

New Paradigm for Virtual Testing in Aerospace
Enzio Maria Provvidone, Vittorio Selmin

Aircraft Process-Oriented Training System – Research and Demonstration Outcomes
Massimo Corazza

TraVis - A Virtual Reality Application For Interactive Simulations In Aerospace And Astronomy
Manuela Marello, Christian Bar

Virtual Reality for Enhanced Urban Design
Mauro Ceconello, Davide Spallazzo

VR/AR Medium and Long Term Application Scenarios in Aerospace Domain
Marinella Ferrino, Domenico Tedone, Benoit Chanclou

13:45 - 15:30 — **Parallel Session III: Sketches and Visions I (Short Papers) – *Chair: Bernd Fröhlich (Uni. Weimar, DE)***

A Low-Cost Laser Based 6 DoF Head Tracker for Usability Application Studies in Virtual Environments
Moritz Vieth, Rainer Herpers, Michael Huelke, Deepak Chhabra

Coperion 3D – A Virtual Factory on the Tabletop
Michael Zöllner, Jens Keil, Johannes Behr, Jan Gillich, Sebastian Gläser, Erich Schöls

Real Time Ray Tracing on Many-Core-Hardware
Iliyan Georgiev, Dmitri Rubinstein, Hilko Hoffmann, Philipp Slusallek

Estimation of Occluded Finger Position in a Camera-Based Tracking System Based on Human Grasping Models
Yutaka Matsuda, Kinya Fujita

WEAR: WEarable Augmented Reality
David De Weerdt, Michel Ilzkovitz, Miguel Casas Sanchez, Tangi Meyer, Luis Arguello

[1] This is an introductory paper to the Haptics session and therefore, not peer-reviewed
[2] Applications I Session papers are not peer-reviewed

Organized by: intuition, MIMOS, ATA, Camera di Commercio Industria Artigianato e Agricoltura di Torino
With the patronage of: Regione Piemonte, Città di Torino

INTUITION 2008 Programme – Tutorials (Oct 6) and Conference Day 1 (Oct 7)

Deforming Virtual Dense Elastic Object with Force Feedback Device PHANToM
Hiroshi Takada, Norihiro Abe, Kinosita Yoshimasa, Shoujie He, Hirokazu Taki

15:30 - 15:45 *Coffee break*

15:45 - 17:30 **Parallel Session IV: Interaction – *Chair: Sabine Coquillart (INRIA, FR)***

The Influence of Position Control and Rate Control on Spatial Perception in Desktop-Based 3D Applications
Jan Hochstrate, Alexander Kulik, André Kunert, Bernd Fröhlich

Wheel-Tracked VR: Walking Through The Virtual World
Javi Boo, Marta Fairén, Jordi Moyés

A System for Aesthetic Shapes Evaluation and Modification Based on Haptic and Auditory Interfaces
Monica Bordegoni, Umberto Cugini

Advanced Interaction Systems to Manage Design Review Inside a Virtual Room
Alice Pignatel, Fausto Brevi

15:45 - 17:30 **Parallel Session V: Applications II – *Chair: Philippe Gravez (CEA, FR)***

Fire in a Storage: From Simulation to Virtual Environments
Andreas Gerndt, Malcolm Hutson, Paul Fung, Carolina Cruz-Neira

Mixed Reality Architecture: Introducing Collaborative VR Tools into Commercial Organisations
Rose Saikayasit, Holger Schnädelbach, Michael Pettitt, Harshada Patel, Tony Glover

Realistic Scene Contrast Reduction Induces Drivers to Slow Down
Paolo Pretto, Astros Chatziastros

Collaborative Editing and Viewing of Interactive 3D Contents on Mobile Devices [3]
Luca Vezzadini, Giuseppe Donvito, Stefano Gasco, Riccardo Corsi

A Virtual Environment for Car Body Trim Assessment in the Automotive Design Process [4]
Alessandro Milite, Arcangelo Truppa, Fabrizio Frascà

[3,4] The paper is invited and thus not peer-reviewed

15:45 - 17:30 **Parallel Session VI: Sketches and Visions II (Short Papers) – *Chair: Fabio Salsedo (PERCRO, IT)***

New Assessment for Old Addictions: Virtual Reality as an Instrument to Assess Alcohol Dependent Patients
Elena Gatti, Cinzia Sacchelli, Rosanna Massari, Tiziana Lops, Riccardo Gatti, Giuseppe Riva

Neural Activity Related to Visual Cues And Saccade Preparation For Operating A Brain Computer Interface
Areti Tzelepi, Ricardo Ron Angevin, Angelos Amditis

Perception of Size in Vehicle Architecture Studies
Andras Kemeny, Emmanuelle Combe, Posselt Javier

Technological and Evaluation Tools for Interaction in Collaborative Virtual Environment
Hrimech Hamid, Merienne Frédéric, Zheng Jian

Avatar Based Physiotherapy Rehabilitation Scheme
Konstantinos Loupos, Italo Braga, Angelos Amditis

Detection of Blood Vessels and Brain Aneurysm Referring to Medical Images of Head
Toshihide Miyagi, Norihiro Abe, Yoshimasa Kinoshita, Shoujie He, Hirokazu Taki

17:30 - 18:15 **Poster Session - Exhibition Rally**

17:30 - 18:50 **VIRMAN Special Session – *Chair: Prof. Satyandra K. Gupta (University of Maryland, USA)***

Use of Simulation in Developing and Characterizing Motion Planning Approaches for Automated Particle Transport Using Optical Tweezers
Ashis Gopal Banerjee, Satyandra K. Gupta

Virtual Worlds as Collaborative Environments for Design and Manufacturing: From Idea to Product
Scott Chase

Haptic Modeling for Virtual Design and Prototyping
Mark B. Colton, Paul A. Theodosis

Use of Virtual Environments in Training Applications: State of the Art Survey
Satyandra K. Gupta, Davinder K. Anand, John E. Brough, Robert A. Kavetsky, Maxim Schwartz

20:30 - *Conference Dinner*

INTUITION 2008 Programme – Conference Day 2 (Oct 8)

Wednesday October 8, 2008

09:00 - 9:45 — **Keynote Speech 3: Convergence of Immersive Media: Virtual Reality Meets Video Games**
Dr Anthony Steed – University College London, UK

9:45 - 10:30 — **Keynote Speech 4: Towards AR-Ready Buildings**
Dr Gudrun Klinker – Technical University of Munich, Germany

10:30 - 10:45 — *Coffee break*

10:45 - 12:15 — **Special Invited Session on Vision and Strategies in VR Technologies and Applications: Industry Perspectives – *Chair: Giuseppe Varalda (CRF, IT)***

12:15 - 12:45 — **Poster Session - Exhibition Rally**

12:45 - 13:45 — *Lunch*

13:45 - 14:30 — **Keynote Speech 5 (VIRMAN *): Virtual Manufacturing – Are We There Yet? The Continuing Journey**
Dr. Sankar Jayaram – Washington State University, USA

14:30 - 16:15 — **Parallel Session VII: Systems – *Chair: Carolina Cruz-Neira (Lite 3D, USA)***

A Global Illumination Rendering Implementation of Web3D Using the Shading Language
Masahiro Sakai, Noriyuki Ichijo, Yoshinori Dobashi, Tsuyoshi Yamamoto

A Low Cost Video See-Through Head Mounted Display for Increased Situation Awareness in an Augmented Environment
Daniel Johansson, Leo J de Vin

Evaluation of Registration Approaches for Industrial Augmented Reality
Katharina Pentenrieder, Christian Bade, Dirk Richter, Fabian Doil, Gudrun Klinker

The USE-VR Platform – A Framework for Interoperabilty Among VR Solutions
Benjamin Mesing, Matthias Vahl, Uwe von Lukas

14:30 - 16:15 — **Parallel Session VIII: European Projects on VR/AR – *Chair: Angelos Amditis (ICCS, GR)***

The following invited European Projects, created in the INTUITION context, are presented:

IMVIS (Bernd Fröhlich - Uni. Weimar, DE)
CATER (Manfred Dangelmaier – FhG-IAO, DE)
MANUVAR (Konstantinos Loupos – ICCS, GR)
COSPACES (Terrence Fernando – USAL, UK)
DIFAC (Claudia Redaelli – ITIA-CNR, IT)

14:30 - 16:15 — **VIRMAN Session I: Virtual Manufacturing Case Studies * – *Chair: Prof. James M. Ritchie (Heriot-Watt University, UK)***

Computing Swept Volumes for Virtual Manufacturing Applications
Huseyin Erdim , Horea T. Ilies

Automated Knowledge Capture in 2D and 3D Design Environments
James M Ritchie, Raymond CW Sung, Heather J Rea, Theodore Lim, Jonathan R Corney, Csaba Salamon, Iris Howley

The Generation of Assembly Process Plans and Associated Gilbreth Motion Study Data
James M. Ritchie, Theo Lim, Raymond S. Sung, Hugo Medellin

Collaborative Virtual Environments for Sharing Product Lifecycle
L. Brayda, L. Taverna, L. Rossi, R. Chellali

Embedding Haptics-Based Virtual Cutting in CAD Systems
Hrishikesh Kate, Sankar Jayaram, Uma Jayaram

14:30 - 16:15 — **VIRMAN Session II: Virtual Manufacturing Case Studies * – *Chair: Prof. Jonathan Corney (University of Strathclyde, UK)***

Virtual Tools and Devices for Automotive Work-Cells Simulation
M. Di Pardo, A. Riccio, M. Allocca, L Talamo

Geometric Reasoning with a Virtual Workforce (Crowdsourcing for CAD/CAM)
P. Jagadeesan, J. Wenzel, J.R. Corney, X.T. Yan, A. Sherlock, W. Regli

A Generic Multimodal Interface for Design and Manufacture Applications
Gheorghe Mogan, Doru Talaba, Florin Garbacia, Tiberiu Butnaru,
Sebastian Sisca, Ciprian Aron

Methods and Tools for Manufacturing Simulation in Aerospace
Marco Alemanni, Giovanni Pricelli

16:15 - 16:30 — **Conference Closure** (Angelos Amditis - ICCS, Giuseppe Varalda - CRF)

* For further information on the VIRMAN programme please visit http://virman08.wordpress.com/programme/

Organized by: intu(i)tion, MIMOS, CAMERA DI COMMERCIO INDUSTRIA ARTIGIANATO E AGRICOLTURA DI TORINO
With the patronage of: REGIONE PIEMONTE, CITTÀ DI TORINO

Sponsors

COMAU — CENTRO RICERCHE FIAT — antycip SIMULATION

ELASIS — IC.IDO The Visual Decision Company — CARCERANO creative engineering

IVECO — projectiondesign HIGH PERFORMANCE PROJECTORS — NoReal.it 3D builders

Exhibitors

Antycip
AR-Tracking
Barco
FhG-IAO + IAT
Haption
ICIDO

Imsys
Istituto Italiano di Tecnologia
Personal Space Technologies
ProjectionDesign
TESS-COM Italia

Organized by: intuition, mimos, AVA, Camera di Commercio Industria Artigianato e Agricoltura di Torino

With the patronage of: Regione Piemonte, Città di Torino